RYAN SLATTERY

MA641 - TIME SERIES ANALYSIS

FINAL PROJECT

SEASONAL TIME SERIES ANALYSIS OF VOYAGER 1 DISTANCE FROM EARTH

DESCRIPTION: AS OF THE TIME I AM WRITING THIS PROJECT, VOYAGER 1 IS THE FARTHEST MAN-MADE SPACECRAFT FROM EARTH. THE SPACECRAFT EXITED THE SOLAR SYSTEM IN 2012 AND CONTINUES TO EXPLORE THE NEW FRONTIER OF INTERSTELLAR SPACE. HOWEVER, A PHENOMENON OBSERVED WITH VOYAGER 1 IS THAT WHILE THE SPACECRAFT TRAVELS AWAY FROM EARTH, THE DISTANCE BETWEEN EARTH AND VOYAGER 1 ACTUALLY DECREASES DURING SOME MONTHS... SO WHY? THIS IS BECAUSE THE EARTH TRAVELS AROUND THE SUN AT A FASTER RATE THAN VOYAGER 1 IS TRAVELLING AWAY FROM EARTH, CAUSING A STEEP INCREASE IN DISTANCE DURING SOME SEASONS AND A SLIGHT DECREASE IN DISTANCE DURING OTHERS. THIS MAKES THE VOYAGER 1 DISTANCE DATASET A PERFECT CANDIDATE FOR SEASONAL TIME SERIES MODELING AND FORECASTING.

```
# LIBRARIES
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(TSA)
```

```
##
## Attaching package: 'TSA'
```

```
## The following objects are masked from 'package:stats':
##
##     acf, arima
```

```
## The following object is masked from 'package:utils':
##
##     tar
```

```
library(forecast)
```

```
## Registered S3 methods overwritten by 'forecast':
##   method       from
##   fitted.Arima TSA
##   plot.Arima   TSA
```

```
library(LSTS)
```

```
##
## Attaching package: 'LSTS'
```

```
## The following object is masked from 'package:TSA':
##
##     periodogram
```

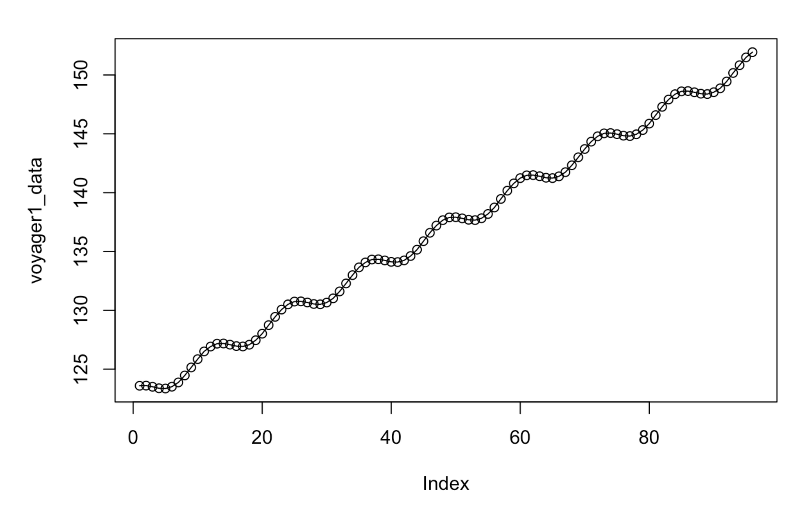
1. READING AND FORMATTING THE DATA

```
# READING THE CSV
voyager1 = read.csv("/Users/ryanslattery/Desktop/Machine\ Learning\ Classes/MA641\ -\ Time\ Series\ Analysis/Proj
ect/Voyager1_Distance_Data.csv", header = FALSE)

# CONVERTING THE DATA TO A NUMERIC VECTOR
voyager1_data = na.omit(as.numeric(unlist(voyager1[2])))
```

```
## Warning in na.omit(as.numeric(unlist(voyager1[2]))): NAs introduced by coercion
```

```
# PLOTTING THE TIME SERIES
plot(voyager1_data, type = "o")
```

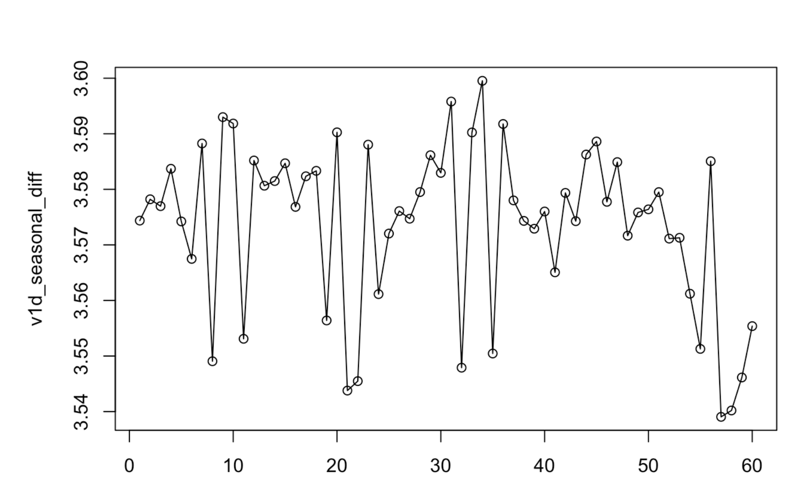



```
# WE WILL BE HOLDING OUT CERTAIN VALUES TO USE LATER FOR OUR FORECASTING. WE WILL ONLY BE OBSERVING 72 OF THE 96
VALUES
forecasting_data = voyager1_data
voyager1_data = voyager1_data[1:72]
```

2. DIFFERENCING TO MAKE THE TIME SERIES STATIONARY (USING A SEASONAL DIFFERENCE AND THEN A REGULAR DIFFERENCE)

```
# PERFORMING THE SEASONAL DIFFERENCE FOR MONTHLY PERIODS
v1d_seasonal_diff = diff(voyager1_data, lag = 12)

# PLOTTING THE SEASONAL DIFFERENCED DATA
plot(v1d_seasonal_diff, type = "o")
```



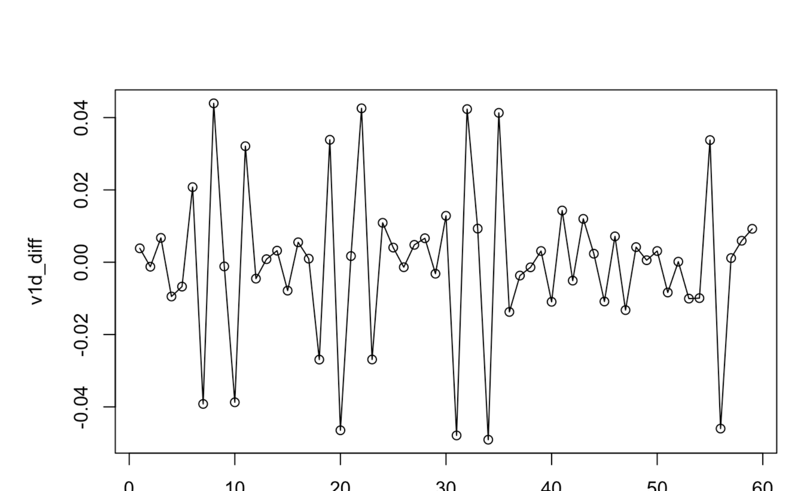

```
# ADF TEST
adf.test(v1d_seasonal_diff)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  v1d_seasonal_diff
## Dickey-Fuller = -1.4612, Lag order = 3, p-value = 0.7925
## alternative hypothesis: stationary
```

THIS TIME SERIES IS NOT STATIONARY SINCE THE AUGMENTED DICKEY-FULLER TEST PRODUCES A P-VALUE GREATER THAN 0.05. MORE DIFFERENCING IS REQUIRED TO MAKE THE SERIES STATIONARY

```
# REGULAR DIFFERENCING
v1d_diff = diff(v1d_seasonal_diff)

# PLOTTING THE TRANSFORMED TIME SERIES
plot(v1d_diff, type = "o")
```




```
# PERFORMING ANOTHER ADF TEST
adf.test(v1d_diff)
```

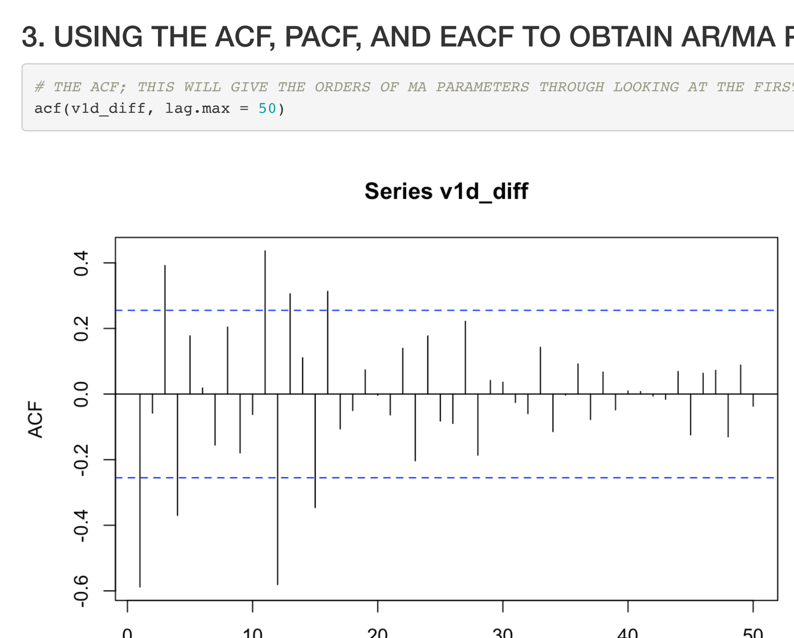
```
## Warning in adf.test(v1d_diff): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  v1d_diff
## Dickey-Fuller = -5.6463, Lag order = 3, p-value = 0.01
## alternative hypothesis: stationary
```

NOW THAT THE P-VALUE IS LESS THAN 0.05, WE CAN OBSERVE THE ACF, PACF, AND EACF TO OBTAIN POTENTIAL ORDERS OF AR/MA MODELS.

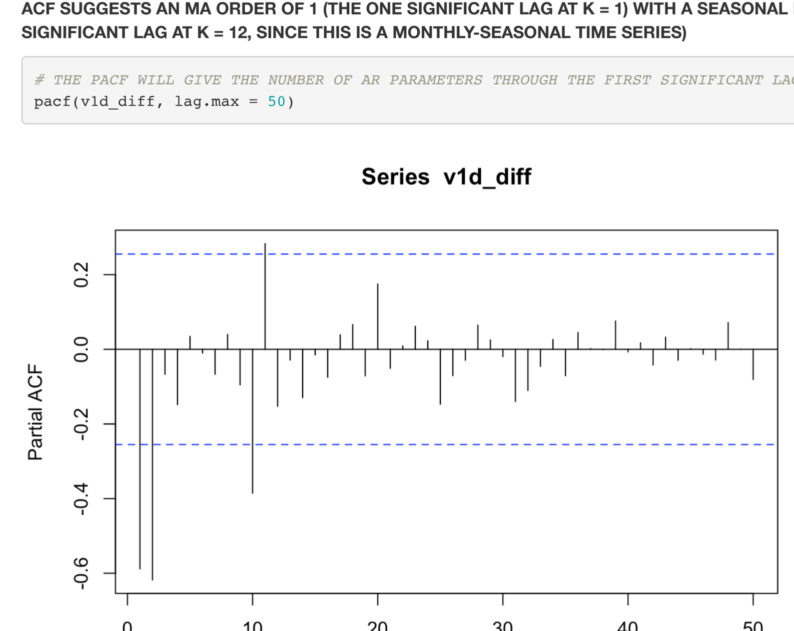
3. USING THE ACF, PACF, AND EACF TO OBTAIN AR/MA PARAMETERS

```
# THE ACF; THIS WILL GIVE THE ORDERS OF MA PARAMETERS THROUGH LOOKING AT THE FIRST SET OF SIGNIFICANT LAGS
acf(v1d_diff, lag.max = 50)
```



ACF SUGGESTS AN MA ORDER OF 1 (THE ONE SIGNIFICANT LAG AT K = 1) WITH A SEASONAL MA ORDER OF 1 (THE ONE SIGNIFICANT LAG AT K = 12, SINCE THIS IS A MONTHLY-SEASONAL TIME SERIES)

```
# THE PACF WILL GIVE THE NUMBER OF AR PARAMETERS THROUGH THE FIRST SIGNIFICANT LAGS.
pacf(v1d_diff, lag.max = 50)
```



PACF SUGGESTS AN AR ORDER OF 2 (FROM THE TWO SIGNIFICANT LAGS AT THE BEGINNING) AND NO SEASONAL AR ORDERS

```
eacf(v1d_diff)
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x o x x o o o o o o x  x  x  o
## 1 x o x o o o o o o o o  x  x  o
## 2 o o o o o o o o o o o  x  o  o
## 3 x o o o o o o o o o o  x  o  o
## 4 x x o o o o o o o o o  x  o  o
## 5 o x o o o o o o o o o  x  o  o
## 6 x o o o o o o o o o o  x  o  o
## 7 x x o o o o o o o o o  o  o  o
```

EACF SUGGESTS THE BEST MODELS COULD BE AR(2), ARMA(2, 1), ARMA(1, 1), AND MA(1).

4. MODEL SELECTION

```
# SARIMA(2, 1, 1)x(0, 1, 1)[12]
v1d_arima1 <- Arima(voyager1_data, order = c(2, 1, 1), seasonal = list(order = c(0, 1, 1), period = 12), method =
"ML")
print(v1d_arima1)
```

```
## Series: voyager1_data
## ARIMA(2,1,1)(0,1,1)[12]
##
## Coefficients:
##           ar1      ar2     ma1     sma1
##       -0.8524  -0.5228  0.0187  -0.7357
## s.e.   0.2128   0.1480  0.2639   0.2305
##
## sigma^2 = 0.003574:  log likelihood = 178.75
## AIC=-347.49   AICc=-346.36   BIC=-337.1
```

```
# SARIMA(1, 1, 1)x(0, 1, 1)[12]
v1d_arima2 <- Arima(voyager1_data, order = c(1, 1, 1), seasonal = list(order = c(0, 1, 1), period = 12), method =
"ML")
print(v1d_arima2)
```

```
## Series: voyager1_data
## ARIMA(1,1,1)(0,1,1)[12]
##
## Coefficients:
##           ar1      ma1     sma1
##       -0.2700  -0.5697  -0.7582
## s.e.   0.1692   0.1451   0.2401
##
## sigma^2 = 0.003523:  log likelihood = 175.64
## AIC=-343.28   AICc=-342.54   BIC=-334.97
```

```
# SARIMA(0, 1, 1)x(0, 1, 1)[12]
v1d_arima3 <- Arima(voyager1_data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1), period = 12), method =
"ML")
print(v1d_arima3)
```

```
## Series: voyager1_data
## ARIMA(0,1,1)(0,1,1)[12]
##
## Coefficients:
##           ma1     sma1
##       -0.7245  -0.7806
## s.e.   0.0971   0.2544
##
## sigma^2 = 0.003464:  log likelihood = 174.6
## AIC=-343.19   AICc=-342.75   BIC=-336.96
```

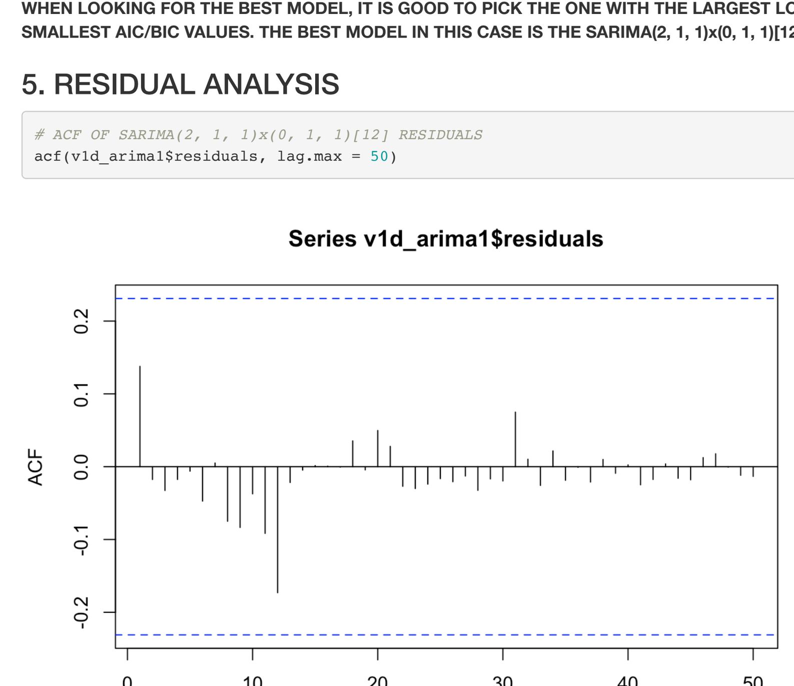
```
# SARIMA(1, 1, 0)x(0, 1, 1)[12]
v1d_arima4 <- Arima(voyager1_data, order = c(1, 1, 0), seasonal = list(order = c(0, 1, 1), period = 12), method =
"ML")
print(v1d_arima4)
```

```
## Series: voyager1_data
## ARIMA(1,1,0)(0,1,1)[12]
##
## Coefficients:
##           ar1     sma1
##       -0.5500  -0.9949
## s.e.   0.1089   0.4340
##
## sigma^2 = 0.003462:  log likelihood = 169.93
## AIC=-333.86   AICc=-333.42   BIC=-327.62
```

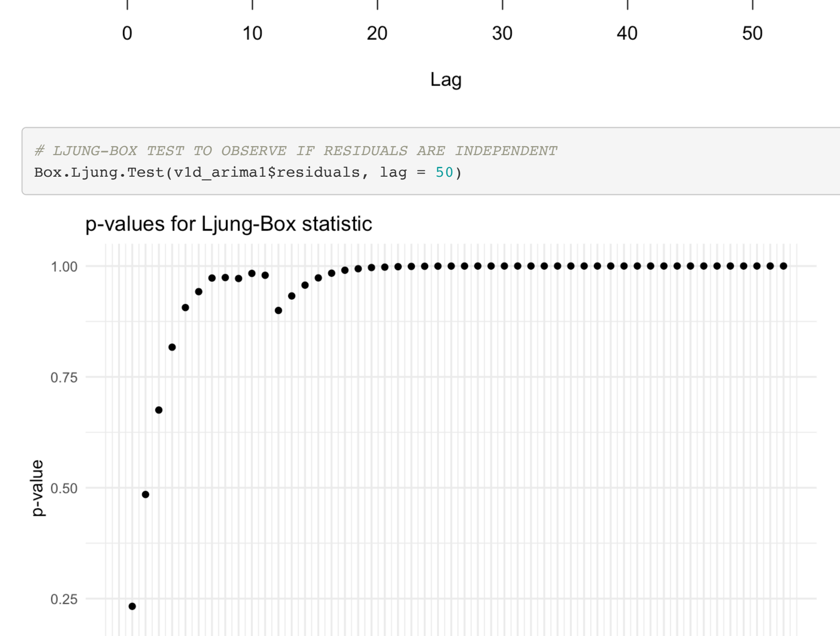
WHEN LOOKING FOR THE BEST MODEL, IT IS GOOD TO PICK THE ONE WITH THE LARGEST LOG-LIKELIHOOD VALUE AND THE SMALLEST AIC/BIC VALUES. THE BEST MODEL IN THIS CASE IS THE SARIMA(2, 1, 1)x(0, 1, 1)[12] MODEL.

5. RESIDUAL ANALYSIS

```
# ACF OF SARIMA(2, 1, 1)x(0, 1, 1)[12] RESIDUALS
acf(v1d_arima1$residuals, lag.max = 50)
```

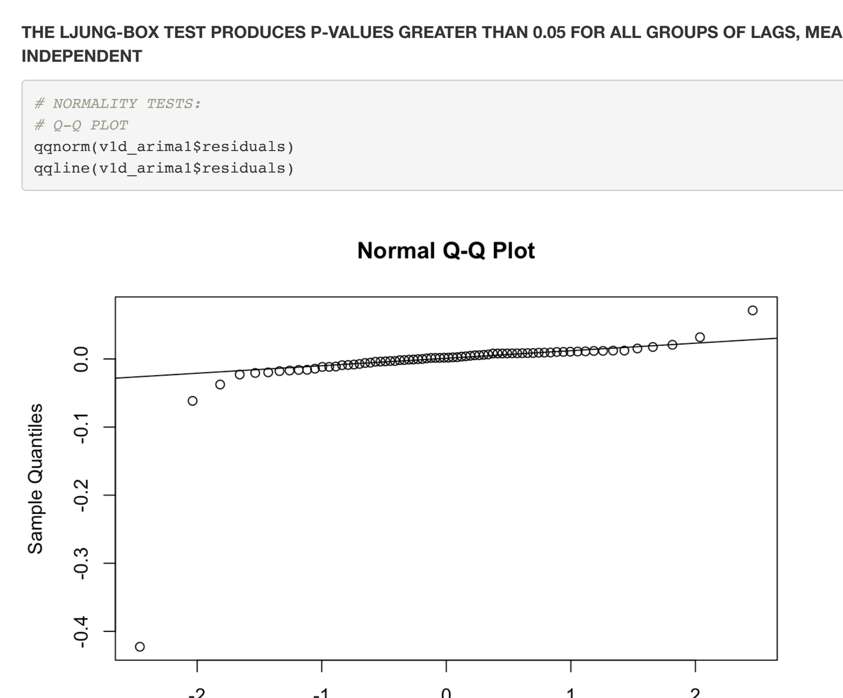



```
# LJUNG-BOX TEST TO OBSERVE IF RESIDUALS ARE INDEPENDENT
Box.Ljung.Test(v1d_arima1$residuals, lag = 50)
```

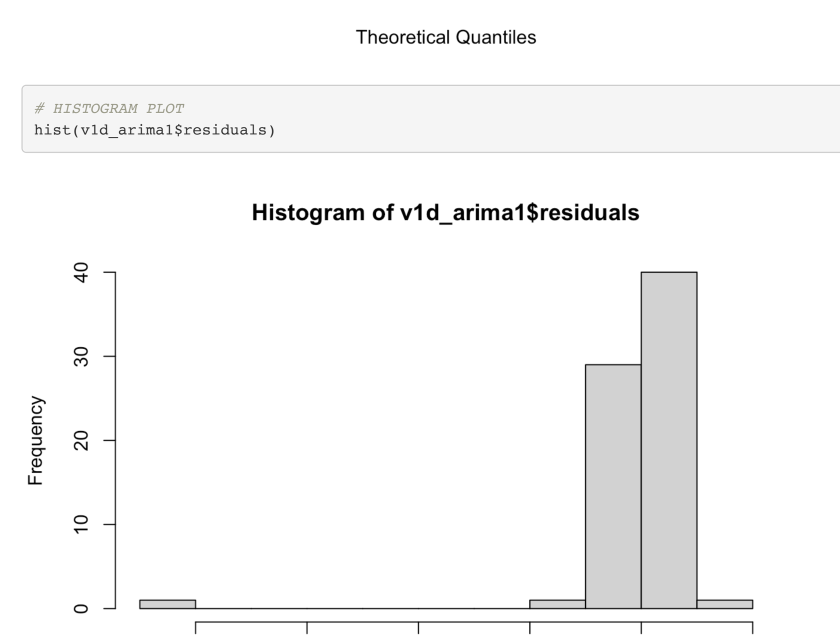


THE LJUNG-BOX TEST PRODUCES P-VALUES GREATER THAN 0.05 FOR ALL GROUPS OF LAGS, MEANING ALL RESIDUALS ARE INDEPENDENT

```
# NORMALITY TESTS:
# Q-Q PLOT
qqnorm(v1d_arima1$residuals)
qqline(v1d_arima1$residuals)
```




```
# HISTOGRAM PLOT
hist(v1d_arima1$residuals)
```




```
# SHAPIRO TEST
shapiro.test(v1d_arima1$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  v1d_arima1$residuals
## W = 0.31294, p-value < 2.2e-16
```

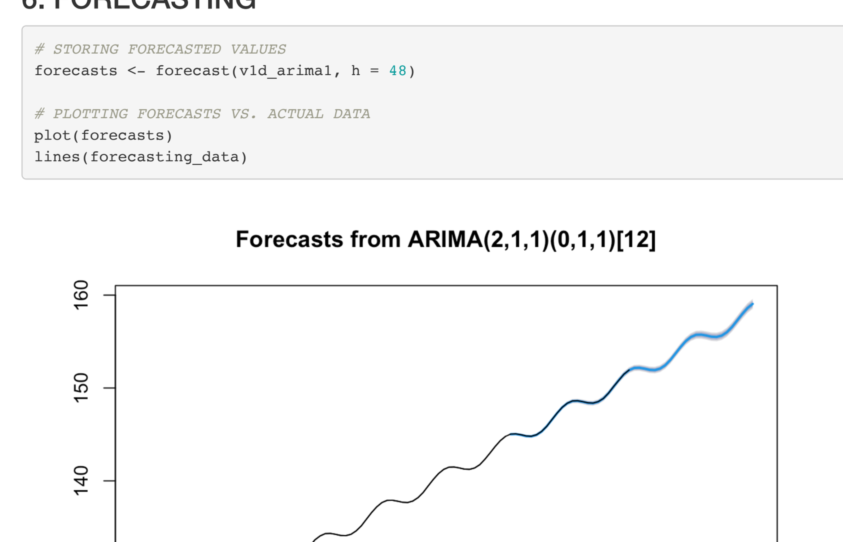
THE HISTOGRAM SEEMS TO RESEMBLE A SOMEWHAT NORMAL DISTRIBUTION, AS THE DATA SEEMS TO BE MOSTLY CENTERED AROUND 0 AND IS ROUGHLY SYMMETRIC. HOWEVER, THERE IS AN OUTLIER AT -0.4 WHICH CAN CAUSE PROBLEMS. AND WHILE THE Q-Q PLOT SEEMS TO BE MOSTLY LINEAR, THE OUTLIER APPEARS TO MAKE THE TAIL HEAVIER AS WELL.

THE SHAPIRO TEST REJECTS NORMALITY WITH A P-VALUE VERY CLOSE TO 0. WHILE IT IS RECOMMENDED TO HAVE NORMAL RESIDUALS FOR UNBIASED FORECASTS, THERE IS NOT MUCH I CAN DO TO MITIGATE THIS ISSUE AND MUST PROCEED WITH WHAT I HAVE.

6. FORECASTING

```
# STORING FORECASTED VALUES
forecasts <- forecast(v1d_arima1, h = 48)

# PLOTTING FORECASTS VS. ACTUAL DATA
plot(forecasts)
lines(forecasting_data)
```



THE FORECASTS OBTAINED IN THE FIGURE ABOVE ARE ACCURATE! THE ACTUAL VALUES FIT IN THE CONFIDENCE INTERVALS OF THE FORECASTED VALUES, AND THE FORECASTS THEMSELVES LOOK LIKE AN EXACT CONTINUATION OF THE DATA!

```
# EVALUATING THE FORECASTS AND ACTUAL DATA WITH MSE
mse = sum((forecasts$mean[1:24] - forecasting_data[73:96])^2) / 24
print(paste0("MEAN SQUARED ERROR FOR SARIMA(2, 1, 1)x(0, 1, 1)12 FORECAST: ", mse))
```

```
## [1] "MEAN SQUARED ERROR FOR SARIMA(2, 1, 1)x(0, 1, 1)12 FORECAST: 0.000363463715109254"
```

THE MSE VALUE OBTAINED IS SMALL, SUGGESTING THAT THE FORECASTED VALUES ARE A GOOD FIT!